APPLICATION

FOR

UNITED STATES LETTERS PATENT

TITLE: REMOTE MIRRORING IN A SWITCHED ENVIRONMENT

APPLICANT: DAVID MEIRI, DAN ARNON, MARK J. HALSTEAD AND
PETER KAMVYSSELIS

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No ___EL224700918US_____

I hereby certify under 37 CFR §1 10 that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, Washington, D.C. 20231.

_____January 23, 2001_____
Date of Deposit

_Samantha Bell_____
Signature

_Samantha Bell_____
Typed or Printed Name of Person Signing Certificate

Remote Mirroring in a Switched Environment

BACKGROUND OF THE INVENTION

5   The invention relates to data storage systems, and
in particular, to data storage systems with remote data
mirroring capability.

        Given the importance of the availability of
information, several techniques have been developed for
10   providing enhanced reliability and availability of data stored
in a data storage system. Once such technique is data
mirroring. In a "mirrored" system, the data stored on one
data storage system is replicated on another data storage
system. Thus, if one or more storage devices on one of the
15   data storage systems fails, or a catastrophic system failure
should occur, the data is readily available in the form of a
mirrored copy from the mirrored data storage system.

        Devices on a data storage system (or source) that
are mirrored on the same remote, data storage system (or
20   target) are referred to as a device group. Likewise, devices
on the target that serve or mirror devices on the same source
are referred to as a device group. Device groups are used,
for example, to insure the consistency of blocks of data too
large to be stored in a single area, during planned or
25   unplanned interruptions such as link failures or planned back-
ups. Device groups typically span a number of physical and
logical volumes, and, in some instances, as when data striping
is used, several device groups may be mapped to different
portions of a physical or logical volume.

30   Typically, the source and target device groups are

arranged as pairs on any two systems and each source/target
device group pair is connected by two dedicated links (e.g.,
ESCON or Fibre Channel links), one for supporting writes
during data backup operations and reads during data recovery
5   operations, and the other for redundancy. Thus, a redundant
arrangement of N data storage systems in which each data
storage system is capable of supporting a mirrored
configuration involving each of the N-1 other remote data
storage systems requires N*(N-1) physical links, or an average
10   of 2*(N-1) physical links per system. This link requirement
becomes impractical when N > 3, as the number of ports on the
data storage system is limited.

SUMMARY OF THE INVENTION

15

        In one aspect of the invention, in a remote data
mirroring arrangement of data storage systems, ports on a data
storage system are connected to ports on other data storage
systems. Each storage system is provided with configuration
20   topology information. From a switch fabric that connects to
ports of all of the data storage systems information
identifying ports of the other data storage systems connected
to the switch fabric is determined. The configuration
topology information and the information obtained from the
25   switch fabric are used to establish a logical link between a
port on the storage system and a second port on a second
storage system so that data residing on a device group
supported by the port and a corresponding, mirrored device
group supported by the second port can be exchanged between
30   the data storage system and the second data storage system.

-2-

In another aspect of the invention, a system includes an arrangement of storage systems each adapted to control at least one group of devices that are supported in a mirrored configuration with a corresponding group of devices

5   controlled by one of the other storage systems. For each device group and corresponding device group, first ports are associated with the device group and second ports are associated with the corresponding device group. The system further includes a switch element adapted to connect one of

10 the first ports to at least one of the second ports so that data may be exchanged between the first and second ports for each device group and corresponding device group.

In yet another aspect of the invention, in a remote, mirrored arrangement of data storage systems, a data storage

15 system includes a port adapted to control at least one device group and a switch element coupled to the port and ports in the other storage systems. The port uses the switch element to link the port to a selected one of the ports controlling a second device group that mirrors the device group controlled

20 by the port.

The advantages of the invention include the following. The replacement of a dedicated one-to-one protocol such as ESCON with a switched protocol reduces the required number of ports from $2(N-1)$ to 2. Such increased connectivity

25 provides for better performance (e.g., load balance) as well as increased redundancy. Also, because each processor on a controller of a data storage system is able to service multiple device groups, a system user can use full remote data mirroring connectivity without having to dedicate processors to

30 supporting that connectivity.

-3-

Other features and advantages of the invention will be apparent from the following detailed description and from the claims.

5
BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a system that includes data storage systems having remote data mirroring capability relative to each other and that are connected to a Fibre
10 Channel (FC) switch fabric.

FIG. 2 is a detailed block diagram of a disk director used to establish logical link connections over the switch fabric for an FC port.

FIGS. 3A-3D are illustrations of configuration
15 topology tables. FIG. 3A is an illustration of a remote systems table. FIG. 3B is an illustration of a device groups table. FIG. 3C is an illustration of a disk directors table. FIG. 3D is an illustration of a logical links table.

FIG. 4 is a format of a World Wide Name field.
20 FIG. 5 is a logical depiction of the system of FIG. 1 with an exemplary configuration of controllers configured to support various device groups.

FIG. 6 is a depiction of a configuration topology graph corresponding to the system shown in FIG. 5.
25 FIG. 7 is a flow diagram of a connection initialization process performed by a processor in a disk director.

FIG. 8 is a flow diagram of a single link discovery process performed by a processor in a disk director.

30

The present invention features a system environment in which data identical to that stored on a data storage system is stored on a geographically remote data storage system. The remote data storage system is used to backup data stored on the data storage system and provide data recovery when the data storage system and its data are lost as a result of malfunction or disaster.

Referring to FIG. 1, a system 10 includes data storage systems 12a, 12b, 12c and 12d. The data storage systems 12a, 12b and 12c are connected to at least one host computer (or host) 14a, 14b and 14c, respectively. The host computer 14 may be, for example, a personal computer, workstation, or the like which may be used by a single user, or a multi-user system. The data storage system 12 receives data and commands from, and delivers data and responses to, the host computer 14.

The data storage systems 12a-12d are mass storage systems having respective controllers 16a-16d, each of which is coupled to pluralities of storage devices (or, simply, devices) shown as disks 18. The controller 16a is coupled to devices 18a, devices 18b and devices 18c. The controller 16b is coupled to devices 18d, devices 18e and devices 18f. The controller 16c is connected to devices 18g, devices 18h and devices 18i. The controller 16d is coupled to devices 18j and devices 18k. Each of the devices 18 is logically divided, in accordance with known techniques, into one or more logical volumes.

Each controller 16 interconnects the host computer

14 and the devices 18, and can be, for example, that made
by EMC and known as the Symmetrix controller. The
controllers 16a-16c receive memory write commands from
respective host computers 14 directly from host buses 20a-
5 20c, respectively, for example, connected and operated in
accordance with a SCSI protocol, and deliver the data
associated with those commands to the appropriate devices
18 over connecting buses 22a, 22b, ...22k. Buses 22 also
preferably operate in accordance with a SCSI protocol. The
10 controllers 16a-16c also receive read requests from the
host computer 14 over host bus 20, and deliver requested
data to the host computer 14, either from a cache memory of
the controller 16 or, if the data is not available in cache
memory, from the devices 18.
15   In a typical configuration, the controller 16 also
connects to a service management console (not shown), which
is used for maintenance and access to the controller 16 and
can be employed to set parameters of the controller 16 as
is well known in the art.
20   Typically, and as indicated above, each of the
devices 18 is configured to store logical volumes (or
devices). There can be a plurality of logical volumes, for
example 4, 8, or more logical volumes, on a physical
device. In a configuration in which multiple copies of a
25 logical volume are maintained, that is, in particular,
where two copies of data are separately stored for a
logical volume, it is said that mirrors are maintained.
(There could be more than two mirrored copies. Typically,

however, mirrored pairs are employed.) The controller 16

can then respond to a read request by reading from either

of the copies stored in the storage devices 18. Mirroring

may exist at both the logical volume and physical device

5   level, or at the logical volume level only. Data mirroring

configurations can occur on the same controller, or on

different controllers, as in the case of remote data

mirroring (or remote data facility, "RDF").

In operation, the host computer 14 sends, as

10   required by the applications it is running, commands to the

data storage system 12 requesting data stored in the

logical volumes or providing data to be written to the

logical volumes. Still referring to FIG. 1, the

controllers 16a-16c include host adapters 24a-24c for

15   facilitating communications with the host computers 14a-c,

respectively. The host computer 14 typically connects to a

port of the host adapter 24 over the SCSI host bus line 20.

The controllers 16a-16d each further include a

global memory 30, shown as global memories 30a-30d,

20   respectively. The host adapter 24 in each controller

connects over at least one system bus 31 to the global

memory 30. Also connected to the global memory system 30

in each of the controllers 16 are disk directors 32, more

specifically, disk directors 32a-32b in the controller 16a,

25   disk directors 32c-32d in the controller 16b, disk

directors 32e-32f in the controller 16c and disk directors

32g-32h in the controller 16d. The disk directors 32

communicates with the host adapter 24 through the global

-7-

memory 30.   Although not shown, the global memory 30 can include a cache memory for storing data, as well as various data structures for maintaining control information and for supporting communications between the hosts 14 and the

5   devices 18.

The disk directors 32 control the storage devices 18.   Each of the disk directors 32a-32h includes a respective one of ports 34a-34h, respectively.   In the described embodiment, the disk directors 32 are installed

10   in the controller 16 in pairs.   Thus, only two disk directors in each of the controllers 16 are shown. However, it will be understood that additional disk directors may be employed by the system 10.

The disk directors in the controllers 16a-16d

15   communicate with the respective global memories 30a-d over dedicated buses 36a-36d, respectively.   During a write operation, the disk director 32 reads data stored in the global memory 30 by the host adapter 24 and writes that data to the appropriate logical volumes.   During a read

20   operation and in response to a read command, the disk director 32 reads data from a logical volume and writes that data to the global memory 30 for later delivery by the host adapter 24 to the requesting host 14.

In the system 10 of FIG. 1, each of controllers 16

25   is connected to a switch fabric 38.   The switch fabric 38 includes a plurality of fabric ports (FP) 40, shown as FP 40a, FP 40b, FP 40c, FP 40c, FP 40d, FP 40e, FP 40f, FP 40g and FP 40h.   Each port in each disk director is connected

to a different one of the fabric ports 40. As shown, ports

34a-34h are coupled to the fabric ports 40a-h,

respectively, over connections or links 42a-42h,

respectively. The switch fabric allows each of the

5   controllers 12 to be connected to other controllers 12 when

the others controllers are serving as remote data storage

systems that provide backup capability in the form of

mirrored storage for the data stored in the data storage

system 12. With respect to any given one of the

10   controllers 16, a "remote" controller or system will be any

other controller 16 that maintains mirrored copies of data

stored on that given controller 16 or stores data for which

the given controller 16 itself maintains mirrored copies.

Thus, any controller 16 and another controller that is a

15   remote controller (and thus provides RDF functionality with

respect to that controller) are said to be in a mirrored

configuration or arrangement. Remote data mirroring

facility and recovery procedures may be performed using

known techniques, such as those described in the above-

20   referenced U.S. Patent No. 5,742,792.

In the described embodiment, the switch fabric 38 is

a Fibre-Channel fabric. However, other one-to-many switch

protocols, e.g., Gigabit Ethernet, can be used.

Other system implementations are contemplated. For

25   example, it will be understood that the data storage

systems 12 need not be limited to one host computer as

shown. For instance the data storage system 12 could be

coupled to more than one host computer. Alternatively, and

as is illustrated with the data storage system 12d, the
data storage system 12 need not be coupled to a host device
at all. Such a system data storage system may be used only
to perform writes in connection with write requests
5   received from one of the other data storage systems 12a-c
so that updated information stored by the host adapter on
that other data storage system is also stored on the remote
data storage system 12d, thereby maintaining the
information stored in the other data storage system, e.g.,
10   data storage system 12a, in a mirrored condition on the
remote data storage system 12d. Also, the host adapter 24
can include a plurality of host adapters, each supporting a
different host bus/host computer. There could be more or
few than the four data storage systems shown in FIG. 1.
15   There could be more than two ports per data storage system.
        Referring to FIG. 2, each disk director 32 includes
a processor 50 coupled to a local, nonvolatile memory (NVM)
52 by an internal bus 54. The processor 50 controls the
overall operations of the disk director 32 and
20   communications with the local memory 52. The nonvolatile
memory 52 stores firmware 56 and parameter data in a
parameter store 58. Also included in the local memory 52
are various data structures 60, i.e., configuration
topology tables 64, which maintain configuration
25   information for the disk director 32 as well as the system
10. The tables 64 include a remote systems table 64a, a
device groups table 64b, a directors (or processors) table
64c and a logical links table 64d. The functionality of

-10-

these data structures will be described later.

Still referring to FIG. 2, the firmware 56 includes a number of processes executed by the processor 50 to control data transfer between the host computer 14 (if present), the global memory 32 and the storage devices 18. In addition, the firmware 56 requires processes to control communications between the port 34 and ports in other controllers via the switch 38. To that end, the firmware 68 is adapted to include a connection initialization process 66 and a single link discovery process 68, as will be described. The firmware 56, data structures 60 and parameter store 58 are read each time the data storage system is initialized. The firmware 56 is copied to the volatile memory 52 at initialization time for subsequent execution by the processor 50.

The various configuration topology tables 64 are illustrated in FIGS. 3A-3D. Referring to FIG. 3A, a remote systems table 64a includes an entry 70 for each controller/data storage system that is involved in a mirrored arrangement with the data storage system. Each entry 70 includes a controller (system) serial number 72; a controller (system) model number 74; and a firmware (code) level identifier 76. Other system-specific information can be included as well.

Referring to FIG. 3B, the device groups table includes an entry for each device group 78, that is, 78a, 78b, …78k, that includes the following: a device group name 80; a pointer to a remote storage system that serves the

device group 82; a list of device group devices 84 in association with characteristics (including RDF parameters) 86. Referring to FIG. 3C, the processor level table 64c includes an entry 90 for each disk director's processor 50,

5   and each entry 90 includes a corresponding name (e.g., WWN, or IP name) 92 and a list of pointers to the device groups supported by the processor 94. Referring to FIG. 3D, the logical links table 96 includes link entries 98. Each includes the following: a link state 100; a pointer to one

10 of the processors 102; and a pointer to one of the device groups 104.

Referring to FIGS. 3A-3D, the tables 64a, 64b and 64c are configured by the system user. The link table 64d is generated by the connection initialization and link

15 discovery processes 66 and 68, respectively (of FIG.2), as will be described.

Referring to FIG. 4, an exemplary format of a World Wide Name (WWN) 106 identified in the field 92 of the processors table 64c is shown. The WWN 106 provides to

20 each processor that is involved in an RDF configuration a unique name. The WWN 106 includes a vendor-specific vendor number field 108 for identifying the manufacturer of the controller 16, a serial number field 110 for identifying a serial number of the controller 16 and a director number

25 112 for identifying the port/processor. In the implementation shown, the lengths of the fields 108, 110 and 112 are 28 bits, 30 bits and 6 bits, respectively.

Referring to FIG. 5, an exemplary logical depiction

of the system 10 (from FIG. 1) is shown. In this example, there are ten device groups, S_1, S_2, S_3, S_4, S_5, T_1, T_2, T_3, T_4 and T_5, which are also indicated by reference numerals 120a, 120b, 120c, 120d, 120e, 122a, 122b, 122c, 122d, and

5   122e, respectively. Five of the device groups, S_1 through S_5, are source device groups, and device groups T_1 through T_5 are target device groups. In the example shown, the controllers are configured for remote data mirroring capability in the following manner: the controller 16a

10   supports device groups S_1, S_2 and T_4; the controller 16b supports device groups S_3, S_5 and T_1; the controller 16c supports device groups S_4, T_3 and T_5; and the controller 16d supports device group T_2. Thus, the devices in source device group S_1 on the controller 16a are mirrored in

15   devices in corresponding target device group T_1 on the controller 16b, the devices in S_2 (in the controller 16a) are mirrored in T_2 (in the controller 16d), and so forth. Each controller has at its disposal 12 different possible logical links. For example, the controller 16b can use its

20   two ports, port 34c and 34d to achieve the following logical links: port 34c to port 34e, port 34c to port 34f, port 34c to 34g, port 34c to port 34h, port 34c to port 34a, port 34c to port 34b, port 34d to port 34e, port 34d to port 34f, port 34d to 34g, port 34d to port 34h, port

25   34d to port 34a and port 34d to port 34b. With the particular configuration illustrated in the figure, a port supporting a device group such as S_3 needs to be in communication with a port for the corresponding device

group, in this case, T_3. As shown in the figure, using the switch architecture and the two ports/processors per controller, a connection between two device groups, e.g., S_3 and T_3, can be achieved with one of four possible logical

5   links: port 34c to port 34e, port 34c to port 34f, port 34d to port 34e and port 34d to port 34f.

Referring now to FIG. 6, an exemplary topology graph that would be maintained by the controller 16b for the system configuration of FIG. 5 is depicted. At the first

10   level, the remote system information 64a includes a representation of controllers 16a (Serial No. 012345678) and 16c (Serial No. 036357760), corresponding to entries in the remote systems table 64a (FIG. 3A). The next level shows device group T_1 pointing to controller 16a because

15   controller 16a maintains S_1, which is the RDF counterpart for the device group T_1. Similarly, device groups S_3 and S_5 point to the controller 16c, as that controller serves these device groups with respective mirrored device groups T_3 and T_5. These device group representations each

20   correspond to an entry in the device group table 64b (FIG. 3B). The next level shows which of the processors (processors associated with ports 34c and 34d) is configured to support which device groups. As represented by the graph (which corresponds to table 64c of FIG. 3C), a

25   processor for one of the directors 32c-32d is configured to support and thus points to device groups T_1 and S_3. The processor for the other one of the directors 32c-32d is configured to support and thus points to S_5. The bottom

level shows the various logical links that are established
(solid lines/arrows) or are desired (dashed lines/arrows).
As shown, logical links are already established for T_1 and
S_3, and the arrows reflect the pointer data stored in the
5   entries of the logical links table 64d (FIG. 3D). For
these established links, the link state is active and the
state value stored in the field 100 of table 64d is '0xff'.

To establish logical links, such as the one between
S_5 and T_5, a port (more specifically, the associated disk
10   director's processor) needs to execute the connection
initialization process 66 and the single link discovery
process 68, as will be described below with reference to
FIGS. 7 and 8, respectively.

Referring to FIG. 7, the connection initialization
15   process 66 (process 66) begins by obtaining from the switch
fabric 40 a list of ports connected to the switch fabric 40
(step 130). Preferably, that list includes the World Wide
Name for each of those ports. The process 66 proceeds to
examine the first item on the list (step 132). The process
20   66 determines if the WWN of the port is greater than or
equal to the WWN of the port on which the process executes,
hereinafter, "our_WWN" (step 134). If it is determined
that the WWN is greater than or equal to our_WWN, the
process 66 determines if there is another item on the list
25   (step 136). If there are no more items on the list, the
process 66 terminates (step 137). If there is another item
to be examined, the process 66 proceeds to the next item on
the list (step 138) and returns to step 134. If the

determination at step 134 indicates that WWN is less than
our_WWN, the process 66 determines if the vendor number is
equal to the vendor number for the port for which the
process executes, or "our_vendor_number" (step 140). If it
5   is not, the process 66 ignores that vendor number and
proceeds to step 136. If the vendor numbers are the same,
the process 66 extracts the serial number and director
number from the WWN (step 142). The process 66 determines
which, if any, device group is served by the WWN (step
10   144). That is, it determines if the serial number extracted
from the WWN matches the serial number pointed to by any
device group. If there is no such match, the process 66
returns to step 136. If there is a match, for each device
group served by the serial number, the process 66 performs
15   the following. The process 66 determines if a logical link
already exists for the device group and serial number (step
146). If a logical link has already been established, the
process 66 returns to step 136. If no logical link exists,
the process 66 adds a link to the link table by setting the
20   state equal to one and adding pointers that point to a
device group served by the matched serial number and the
processor that points to that device group (step 148). The
process 66 initiates the single link discovery process 68
to establish a link to the remote processor identified by
25   the WWN director number (step 150). The process 66 then
returns to step 136. Thus, the process 66 performs steps
134 through 150 for each port represented on the list. The
process 66 is performed as a background process and is

- 16 -

repeated in accordance with a timing parameter set in the parameter store 58 (FIG. 2).

Referring to FIG. 8, a detailed flow of the single link discovery process 68 (process 68) is shown. The process 68 initiates a login process for the port that is initiating the link discovery (initiator port) and the fabric (step 160). In the described embodiment for a FC switch, the login is performed using known FC login techniques. The process 68 of the initiator port initiates contact with the remote (responder) port by sending a synchronization message to the responder port (step 162). Synchronization may be achieved in this manner by resetting the responder port. If a timeout occurs prior to receiving a response or the link attempt is rejected by the fabric because the link from the fabric to the responder port is has failed (step 164), the process 68 retries synchronization after a predetermined retry delay has occurred (step 166). Otherwise, a response is received from the responder port and the synchronization message causes the responder port to perform its own FC login, as well as create a link entry for the port with a link state value of one as part of its own link discovery process (step 168). The initiator port reads the configuration information for the device group pointed to by the remote port and compares it to the configuration for the device group (step 170). It provides the device group to the responder port, which causes the responder group to determine if it supports the device group by reading the

-17-

configuration data for the initiator port's device group and comparing it to configuration information stored for the remote device group (step 172). If the process does not detect or receive indication of a mismatch, that is the

5 link discovery is successful (step 174), it causes the initiator port to advance its link state to a an '0xff' value (step 176). Alternatively, if the process 68 determines that link discovery is unsuccessful due to an RDF mismatch/rejection, the process 68 caches the

10 information about the failure (notes the WWN of the remote port) so that the process 68 can avoid repeating attempts to establish a connection that cannot be established due to configuration limitations or mismatches (step 178).

Thus, during a single link discovery, the process 68

15 causes the state machines in both the initiator and responder ports to attempt to advance their state from '1' to '0xff' with four possible results: 'ff' (link established); timeout (fail) with retry; "link down" rejection; or RDF rejection. The link is successfully

20 established when both state machines have advanced the link state to 'ff'.

It will be appreciated that numerous other modifications may be made in connection with the invention. The controllers 16 may further support a mirror service policy

25 which identifies which disk directors in conjunction with the storage devices that they control are primarily responsible for satisfying requests from a host computer. This policy could be fixed at the beginning of a system set up, taking

into account the expected loads. Preferably, however, the
mirror service policy implementation is dynamic. That is, the
mirror service policy can be modified, periodically, in
response to statistics describing the nature of the read and

5   write requests to the data storage system 12, to change the
mirror service policy during the operation of the data storage
system 12. An example of a dynamic mirror service policy
(DMSP) is described in U.S. Patent No. 6,112,257.

Additions, subtractions, and other modifications of

10   the preferred embodiments of the invention will be apparent to
those practiced in this field and are within the scope of the
following claims.

What is claimed is: